

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: : Confirmation No.: 2283
:
Cosatto et al. : Attorney Ref.: 2000-0042-CON
:
Serial No.: 10/662,550 : Art Unit: 2628
:
Filed: September 15, 2003 : Examiner: Daniel F. Hajnik
:
FOR: AUDIO-VISUAL SELECTION PROCESS FOR THE SYNTHESIS OF PHOTO-
REALISTIC TALKING-HEAD ANIMATIONS

AFFIDAVIT UNDER 37 CFR 1.131

Honorable Commissioner of Patents & Trademarks

Alexandria, VA 22314

Dear Sir:

We, Eric Cosatto, Hans Peter Graf, Gerasimos Potamianos, and Juergen Schroeter, being
duly sworn, depose and state:


1. We are the coinventors of claims 22-25, 27-32, and 34-35 of the above-identified patent application.
2. The above-identified patent application was filed 15 September 2003.
3. The above-identified application claims priority to a parent application 2000-0042, now issued as U.S. patent 6,654,018, filed 29 March 2001.

4. Exhibit #1 is an internal AT&T memo from Hans Peter Graf, coinventor, to Thomas Restaino, General Attorney AT&T Corporate Affairs, dated 19 January 1999, and includes a draft technical paper disclosing the invention.
5. The technical paper in Exhibit #1 shows that we conceived the idea of an audio-visual selection process for synthesizing photo-realistic talking head animations as described and claimed in our application no later than 19 January 1999. The details of our invention are outlined in Exhibit #1, pages 3-12, under the paper titled "Photo-Realistic Talking-Heads from Image Samples".
6. Exhibit #2 is an internal AT&T document to process new invention submissions.
7. Exhibit #2 indicates that IDS No. 2000-0042, the disclosure for the parent application of the present application, was approved for filing on 17 March 2000 by the Multimedia Intellectual Property Review (IPR) Team of AT&T Corp.
8. Thomas Restaino (TAR), General Attorney of AT&T Corporate Affairs, or his secretary (amf) approved and signed Exhibit #2 and sent the disclosure to Ann Taylor, Outside Counsel Coordinator, on 4 April 2000 with instructions to assign the disclosure to outside counsel Wendy Koba to prepare and file the patent application.
9. Exhibit #3 shows the front and back of a mail-in disclosure receipt sent from Wendy Koba to AT&T. The receipt was stamped to indicate that AT&T Corp. received the receipt from Wendy Koba on 12 April 2000.
10. The application was filed 29 March 2001 and assigned Appl. No. 09/820,396.

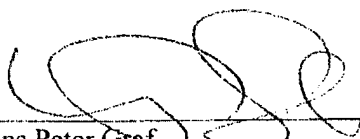
Application/Control Number: 10/662,550
Art Unit: 2628

Docket No.: 2000-0042-CON

All the statements made herein are true and are made on information believed to be true; and further these statements are made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. 1001 and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.


Eric Cosatto

11/12/2008
Date


Hans Peter Graf

11/14/2008
Date

Gerasimos Potamianos

Date

Juergen Schroeter

Date

Application/Control Number: 10/662,550
Art Unit: 2628

Docket No.: 2000-0042-CON

All the statements made herein are true and are made on information believed to be true; and further these statements are made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. 1001 and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Eric Cosatto

Date

Hans Peter Graf

Date


Gerasimos Potamianos

11/13/08

Date

Juergen Schroeter

Date

Application/Control Number: 10/662,550
Art Unit: 2628

Docket No.: 2000-0042-CON

All the statements made herein are true and are made on information believed to be true; and further these statements are made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. 1001 and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Eric Cosatto

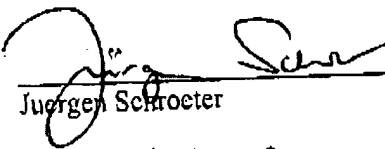
Date

Hans Pcter Graf

Date

Gerasimos Potamianos

Date


Juergen Schroeter

11/7/2008

Date

Application/Control Number: 10/662,550
Art Unit: 2628

Docket No.: 2000-0042-CON

EXHIBIT #1

Hans Peter Graf
AT&T Labs Research
Room 3-134
100 Schulz Drive
Red Bank, NJ 07701
(732) 345 3338; Fax: (732) 345 3031
hpg@research.att.com

2000-0042

1/26/99

Tom Restaino
General Attorney
AT&T Corporate Affairs
150 Allen Rd
Liberty Corner, NJ 07938

January 19, 1999

Re: Patent submission

Tom,

The attached paper will be submitted to a conference with a publication date of early August 1999 - the official release forms will come in a few days. After discussions with several people we think that the two ideas mentioned below should be covered by patent(s) and send you here the information that you can evaluate them.

Thank you



Hans Peter Graf

The attached paper contains two aspects that should be considered for coverage by patents.

Combining 3D modeling with sample-based rendering

We developed a new technique for generating photo-realistic images and animations by combining 3D modeling with sample-based rendering (described mainly in sections 3 and 5).

Synthesizing scenes that look like real photographs is an extremely difficult problem and is one of the hottest research topics in computer graphics today. The main approach is to refine 3D models to the point where they look very similar to real-world objects. Yet compute requirements are growing exponentially as finer and finer levels of detail need to be modeled. Recently sample-based rendering is gaining interest as an alternative for generating photorealistic scenes. This technique starts from photos and synthesizes new scenes by integrating fragments of these photos. The main drawbacks of sample-based rendering are the large number of data that need to be recorded and a lack of flexibility in generating new scenes.

Our new approach overcomes the problems of previous techniques by combining the flexibility of 3D modeling with the photo-realism of sample-based rendering. The paper describes this technique for generating talking heads, yet the concept is not limited to heads and faces, but is very general and applicable to any 3D object.

Perceptual parameters of texture maps

One of the main problems for generating photo-realistic scenes is to find texture maps appropriate for integration into a new scene. To solve this problem we developed a new technique for describing the perceptual appearance of texture maps (briefly described in 5.3). We can now search photographs automatically for segments with a certain appearance and generate databases of texture maps. For generating a new scene one specifies the desired appearance with a few perceptual parameters and can recall the texture maps from a database.

The new technique applies a combination of geometric characterizations plus filtering and morphological operations to describe the appearance of a texture map.

The two techniques mentioned above have many potential applications in computer graphics, animation, and image coding. The talking-head animations described in the paper are just one example. Another potential application is image compression, which relies more and more on symbolic descriptions of scenes (for example, in MPEG4 faces can be encoded in this way). The decoder then synthesizes the image from a few transmitted parameters. Such techniques rely on generating photorealistic scenes.

Photo-Realistic Talking-Heads from Image Samples

Eric Cosatto & Hans Peter Graf

AT&T Labs-Research, 100 Schulz Drive, Room 3-124,134, Red Bank, NJ 07701-7033, USA [eric,hpg]@research.att.com

Abstract

This paper describes a system for creating a photo-realistic model of the head that can be animated and lip-synched from a string of phonemes. Combined with a state-of-the-art text-to-speech synthesizer, it generates video animations of talking heads that closely resemble real people.

To obtain a naturally looking head, we choose a 'data-driven' approach. We record a talking person and apply image recognition to extract automatically bitmaps of facial parts. These bitmaps are normalized and parameterized before being entered into a database. For the synthesis, we start from a phoneme string and calculate motion trajectories for all the facial parts and the whole head. These trajectories provide the parameters for selecting the proper bitmaps from the database. Smoothing and blending is applied to these 'strings' of bitmaps to eliminate hard transitions and create a seamless animation for each facial part.

A simple 3D model of the head guides the blending of the bitmaps into a whole head with a given pose. This model follows only the rigid movements of the head, and, unlike traditional 3D models, it does not deform with plastic deformations of the face. It only serves as a guide for extracting and combining bitmaps from/to an image. For example, the mouth area is modeled with six planes, one for the lips, two for the cheeks and three for the jaw. Because the recorded bitmaps only incur minor deformations, due to the slight warps associated with rotations, their original appearance is preserved. The result is a talking head that resembles very closely the person who was originally recorded.

Talking-head animations of this type are useful as a front-end for agents and avatars in such applications as virtual operators, help desks, educational and expert systems.

Keywords

Facial animation, computer vision, image-based rendering.

1 INTRODUCTION

Animated characters and in particular talking heads are playing an increasingly important role in computer interfaces. An animated talking head attracts immediately the attention of a user, can make a task more engaging and adds entertainment value to an application. Seeing a face makes many people feel more comfortable interacting with a computer. For learning tasks several researchers report that animated characters can increase the attention span of the user, and hence improve learning results. When used as avatars, lively talking heads can make an encounter in a virtual world more engaging. Today such heads are usually either recorded video clips of real people or cartoon

characters lip-synching synthesized text.

Often a cartoon character or robot-like face may do, yet we respond to nothing as strongly as to a real face. For an educational program, for example, a real face is preferable. A cartoon face is associated with entertainment, not to be taken too seriously. An animated face of a competent teacher, on the other hand, can create an atmosphere conducive to learning and therefore increase the impact of such educational software.

Generating animated talking heads that look like real people is a very challenging task, and so far all synthesized heads are still far from reaching this goal. To be considered natural a face has to be not just photo-realistic in appearance, but must also exhibit realistic head movements, emotional expressions, and proper plastic deformations of the lips synchronized with the speech. We are trained since birth to recognize faces and facial expressions and therefore are highly sensitive to the slightest imperfections in a talking face.

Instead of modeling a human head in minute detail, we start from photographic images of a person's face and generate animated sequences from these samples. This sample-based modeling approach preserves a high level of detail in the appearance of the face. By recording real movements of the head and of the lips and reusing them for the synthesis we obtain a model that is able to produce realistic lip and head movements, as well as emotional expressions.

Section 3 defines how the head and its facial parts are modeled and the process of capturing sample data. In order to capture accurately realistic speech postures, we have subjects speak short text sequences in front of the camera. A face recognition system then analyzes automatically this video footage and selects the proper samples as described in section 4. Section 5 presents the process of extracting bitmaps from video frames and how they are normalized and parameterized for an easy access in a database. Finally, the synthesis of the talking-head animation driven by a string of phonemes is described in section 6.

2 PREVIOUS WORK

Many different systems exist for modeling the human head [8], achieving various degrees of photo-realism and flexibility, but relatively few have demonstrated a complete talking-head functionality.

2.1 3D head modeling

Most approaches use 3D meshes to model in fine detail the shape of the head [11][12]. These models are created using advanced 3D scanning techniques such as a CyberWare range scanner [19] or are adapted from generic models using either optical flow constraints [14] or facial features labeling [3][4]. Some of them include information on how to move vertices

according to physical properties of the skin and the underlying muscles [19]. To obtain a natural appearance they typically use images of a person that are texture-mapped onto the 3D model. Yet, when plastic deformations occur, the texture images are distorted resulting in visible artifacts. Another difficult problem is modeling of hair and such surface features as grooves and wrinkles. These are important for the appearance of a face and yet are only marginally, if at all, modeled by most of these systems. The incredible complexity of plastic deformations in talking faces makes precise modeling extremely difficult. Simplifications of the models result in unnatural appearances and synthetic looking faces.

2.2 Morphing 2D views

An alternative approach is based on morphing between 2D images. These techniques can produce photo-realistic images of new shapes by interpolating between two existing shapes. Morphing of a face requires precise specifications of the displacements of many points in order to guarantee that the results look like real faces. Most techniques therefore rely on a manual specification of the morph parameters [16]. Beymer et al. [15] and Bichsel [17] have proposed image analysis methods where the morph parameters are determined automatically, based on optical flow. While this approach gives an elegant solution to generating new views from a set of reference images, one still has to find the proper reference images. Moreover, since the techniques are based on 2D images the range of expressions and movements they can produce is rather limited.

2.3 Sample-Based synthesis

Recently there has been a surge of interest in sample-based techniques (also referred to as data-driven) for synthesizing photo-realistic scenes. These techniques generally start by observing and collecting samples that are representative of a signal we wish to model. The samples are then parameterized so that they can be recalled at synthesis time. Typically samples are processed as little as possible to avoid distortions. One of the early successful applications of this concept is QuickTime VR® [20]. This system allows panoramic viewing of scenes as well as examining objects from all angles. Samples are parameterized by the direction from where they were recorded and stored in a two-dimensional database.

Recently other researchers explored ways of sampling both, texture and 3D geometry of faces [3][4], producing impressive animations of facial expressions. These systems use multiple cameras or facial markers to derive the 3D geometry and texture of the face in each frame of video sequences. Deriving the exact geometry of such details as grooves, wrinkles, lips, and tongue as they undergo plastic deformations remains, however, difficult. Extensive manual measuring in the images is required, resulting in a labor intensive capture process. Textures are processed extensively to match the underlying 3D model and may lose some of their natural appearance. These systems have not yet been demonstrated for speech reproduction.

2.4 Talking-Head systems

A talking-head synthesis technique based on recorded samples that are selected automatically has been proposed by Bregler et al. [13]. This system can produce videos of real people uttering text they never actually said. It uses video snippets of in-

phones (3 subsequent phonemes) as samples. Since these video snippets are parameterized with the phoneme sequence, the resulting database is very large. Moreover, this parameterization can only be applied to the mouth area, precluding the use of other facial parts such as eyes and eyebrows that are carrying important conversational cues.

Ezzat et al. [5] have demonstrated a sample-based talking head system that uses morphing to generate intermediate appearances of mouth shapes from a very small set of manually selected mouth samples. While morphing generates smooth transitions between mouth samples, this system does not model the whole head and does not synthesize head movements and facial expressions. Cosatto et al. [6] presented a sample-based talking head that uses several layers of 2D bit-planes as a model. Neither facial parts nor the whole head are modeled in 3D and therefore the system is limited in what new expressions and movements it can synthesize.

3 MODEL

3.1 Definition

In defining our model of the head, we attempt to combine the flexibility of 3D models with the realism of images. A key problem with sample-based techniques is to control the number of image samples that need to be recorded and stored. A face's appearance changes due to talking, emotional expressions and head orientation, leading to a combinatorial explosion in the number of different appearances. To keep the number of samples at a manageable level we divide the face into a hierarchy of parts and model each part independently. This results in a compact model that can create animations with head movements, speech articulation and different emotional expressions.

Our face model is defined as follows:

1. Hierarchy of parts: The head is separated into a 'base face' (Figure 1a) and a number of facial parts. The base face covers the area of the whole face serving as a substrate onto which the facial parts are integrated. The facial parts are: mouth with cheeks, jaw, eyes, and forehead with eyebrows (Figure 1b). Nose and ears are not modeled separately, but are part of the base face.
2. 3D model: The shape of each facial part is approximated with a small number of planes. This set of planes is used as a guide to map the facial parts onto the base face in a given pose (Figure 1c,d). The positions and orientations of these planes follow the movements of the head, yet their shapes remain constant, even when the corresponding facial parts undergo plastic deformations. Hence, a model plane is more like a local window onto which a facial part is projected than a polygon of a traditional 3D model.
3. Sample bitmaps: For each facial part, sample bitmaps are recorded that cover the range of possible appearances produced by plastic deformations. No separate bitmaps are recorded to account for different head orientations. For the base face, bitmap samples are recorded with the head in different orientations. The range of head rotations we consider at the moment is $\pm 15^\circ$.

There is no unique way of decomposing a face into parts, and no part of the face is truly independent from the rest. Muscles and skin are highly elastic and tend to spread a deformation in one place across a large part of the whole face. The decomposition described here was chosen after studying, how

facial expressions are generated by humans [25], and how they are depicted by artists and cartoonists [24].

To generate a face with a certain mouth shape and emotional expression, the proper bitmaps are chosen for each of the facial parts. The head orientation is known from the base face, so that we can project the bitmaps onto the base face using simple warping [23]. This operation is similar to traditional texture mapping. The difference with traditional 3D modeling techniques is that for plastic deformations we select different bitmaps, rather than trying to squeeze one single bitmap into any new shape. Only rigid movements such as rotation and translation of the whole head plus the rotation of the jaw are modeled. The bitmaps of the facial parts are integrated into the base face with proper feathering (alpha-blending at the edges), so that they blend smoothly into the base face, without introducing artifacts (Figure 1f).

We consider here only a limited range of frontal views that are typical for movements during spontaneous speech. Then we do not need bitmaps recorded with different head orientations for the facial parts. Empirical studies showed that for a range of $\pm 15^\circ$ of rotation, warping does not introduce serious distortions in the bitmaps that would be considered unnatural. We limit the discussion here to a range that can be covered with a single set of bitmaps and model planes as shown in figure 1. The model can, however, be adapted to cover a wide range of orientations. Sample bitmaps have to be recorded from different angles, and the planes of the model need to be adjusted.

3.2 Capturing sample bitmaps

A head model is instantiated in two steps. First a few measurements are made on the subject's face to determine its geometry, namely the relative positions of eye corners, nostrils, mouth corners and the bottom of the chin. Using these measurements, the model planes are adapted for each facial part. Since this is done only once, there is little incentive to automate it. Techniques exist, such as the one described in [14] that can adapt a generic head model from video sequences showing head movements. This may be useful if only video footage exists without the person being present.

Once the 3D model is defined, each face part is populated with bitmaps representative of its appearances. A person is recorded while speaking freely a few short sentences, to get all the different mouth shapes. For the examples shown here the lady spoke 14 phrases, each two to three seconds long. We try to keep the capture process as simple and non-intrusive as possible, since we are interested in capturing the typical head movements during speech as well as special mimics and unique ways of articulating words. In particular, we avoid any head restraints or forced pose, such as requiring the subject to watch constantly in a given direction. Guenter et al. [3] present a sophisticated technique that involves gluing dozens of fluorescent dots on the subject's face. Later, image processing is able to remove the dots. Thanks to robust recognition algorithms (see section 4) we do not need any special markers on the subject's face, but rather exploit the natural richness of features of the face. We also avoid the need of multiple cameras. Knowing the positions of a few points in the face allows recovery of the head pose using techniques described in section 4.

Lip movements can be extremely fast, which may cause blurry images, when the frame rate is not high enough. Recording 60 fields per second instead of 30 frames, or using a shutter can

solve this problem without having to resort to expensive cameras. We adjust luminance and hue of the facial parts and the base face, so that they will blend seamlessly. By making sure that the illumination is reasonably homogeneous one can avoid excessive color corrections that may introduce artifacts. Moreover, having a background of uniform and neutral color makes finding the location of the head easy. We currently capture frames of 560x480 pixels in size with the head being about four fifths of this height, ensuring a high level of fidelity in rendering details of facial features, skin and hair.

Quite some effort has gone into developing the whole system in a way that the capturing process remains easy and cheap. Eventually the system should be usable outside the lab by relatively unskilled personnel, or even at home by the user himself. The final goal is to have an easy procedure where you can quickly produce an animated head of yourself.

4 RECOGNITION

Sample-based synthesis of talking heads depends on a reliable and accurate recognition of the face and the positions of the facial features. Without an automatic technique for extracting and normalizing facial features, a manual segmentation of the images has to be done. Considering that we need samples of all lip shapes, of different head orientations and of several emotional expressions, thousands of images have to be searched for the proper shapes. If we also want to analyze the lip movements during transitions between phonemes, we have to analyze hundreds of thousands of images. Clearly, it is not feasible to do such a task manually.

The main challenge for the face recognition system is the high precision with which the facial features have to be located. An error as small as a single pixel in the position of a feature distorts the pose estimation of the head noticeably. To achieve such a high precision our analysis proceeds in three steps, each with an increased resolution. The first step finds a coarse outline of the head plus estimates of the positions of the major facial features. In the second step the areas around the mouth, the nostrils and the eyes are analyzed in more detail. The third step, finally, zooms in on specific areas of facial features, such as the corners of the eyes, of the mouth and of the eyebrows and measures their positions with high accuracy.

4.1 Locating the face

In a first step the whole image is searched for the presence of heads, and their locations are determined. Each frame is analyzed with two different algorithms. The first type of analysis is a color segmentation to find the areas with skin colors and colors representative of the hair.

The second type of analysis segments the image based on textures and shapes. This analysis uses only the luminance of the image. First, the image is filtered with a band-pass filter, removing the highest and lowest spatial frequencies (figure 2a). Then a morphological operation followed by adaptive thresholding results in a binary image where areas of facial features are marked with blobs of black pixels (figure 2b).

The color analysis as well as the texture analysis produce sets of features. Combinations of these features are evaluated with classifiers, testing their shapes and relative positions. For example, an area marked by the color analysis as a candidate of a face area is combined with candidates of eye areas produced by the texture analysis. If relative sizes and positions match closely

those of a reference face, this combination is evaluated further and combined with other features. Otherwise it is discarded.

In order to save computation time, the analysis starts with simple representations, and only if the result is not satisfactory, a more complex representation is used. For example, when a classifier tries to determine whether three features represent two eyes and a mouth, it takes in a first pass only the center of mass of each feature into account and measures their relative positions. If the results are ambiguous, the analysis is repeated looking also at the shape of each feature, using the outlines of each connected component in the image.

This bottom-up approach of evaluating combinations of features produces reliably and quickly the location of the head as well as the positions of the major facial features (figure 2c). In the videos used for extracting samples, there is usually only one person present and the lighting is fairly uniform. Moreover, the background is static with little texture. Locating the head in images of such quality is rather easy and can even be done at a strongly reduced resolution without losing reliability. Typically, images are down-sampled to a quarter or a ninth of the original size for this analysis in order to speed it up.

4.2 Locating facial features

Finding the exact dimensions of the facial features is more challenging, since the person being recorded is moving the head and is changing facial expressions while speaking. This can lead to great variations in the appearance of a facial feature and can also affect the lighting conditions. For example, during a nod a shadow may fall over the eyes. Therefore, the analysis described above does not always produce accurate results for all facial features and we need to analyze further the areas around eyes, mouth, and the lower end of the nose.

The algorithm proceeds by analyzing the color space, periodically retraining it with a small number of frames. For example, the area around the mouth is cut out from five frames and with a leader-clustering algorithm the most prominent colors in the area are identified. By analyzing the shapes of the color segments we can assign the colors to different parts, such as the mouth cavity, the teeth and the lips (figure 2d, 2e, 2f). By repeating the color calibration periodically, we keep track of changes in the appearances of the facial features.

The texture analysis is also adapted to the particular facial feature under investigation by adjusting the filter parameters to the size and shape of a feature. In this way, the combination of texture and color analysis produces reliable measurements of the positions and outlines of the facial features.

Errors made by the system are of two types. The first type is a complete failure to identify a facial feature and the second type is inaccuracy in the measurements. A failure to identify a feature happens in about 1% of the frames, mostly when the head moves over a wider range than what was seen in the training images. The accuracy achieved for the dimensions of the mouth are typically ± 2 pixels (standard deviation), where the width of the mouth is around 100 pixels. More details on this face analysis system can be found in [18]. It has been tested in a large number of lip reading experiments, analyzing lip shapes of 50 different people pronouncing over 5,000 utterances, recorded under varying lighting conditions [10].

4.3 High accuracy feature points

For measuring the head pose, a few points in the face have to

be measured with high accuracy, preferably with an error of less than one pixel. The techniques described above tend to produce variations of, for example, ± 2 pixels for the eye corners. Filtering over time can improve these errors significantly, yet a more precise measurement is still preferable.

We therefore add a third level of analysis to measure a few feature points with the highest accuracy. From a training set of 300 frames a few representative examples of one feature point are selected. For example, for measuring the position of the left lip corner, nine examples are selected (figure 2i). These samples are chosen based on the dimensions of the mouth (figure 2h). This means that the training procedure selects mouth images with three different widths and three different heights. From those images the areas around the left corner are cut out. For analyzing a new image, one of these sample images is chosen, namely the one where the mouth width and height are most similar, and this kernel is scanned over an area around the left half of the mouth.

To measure the similarity between the kernel and the area being analyzed, both are filtered with a high-pass filter before multiplying them pixel by pixel (figure 2j). This convolution identifies very precisely where a feature point is located (figure 2k). The standard deviation of the measurements is typically less than one pixel for the eye corners and filtering over time reduces the error to less than 0.5 pixels.

The time required for this operation scales with the kernel size times the size of the analyzed area. We found empirically that a kernel size of 20x20 pixels provides adequate robustness and analyzing an area of 100x100 pixel takes less than 100 ms on a 300MHz PC.

The features we measure are mouth, nostrils, eyes and eyebrows. Knowing the positions and shapes of these features is sufficient to identify visemes of the mouth and the most prominent emotional expressions. Sometimes the interior of the mouth is also analyzed to get a better measure of lip protrusion and stress put on the lips.

4.4 Pose estimation

We apply a pose estimation technique reported in [21], using six feature points in the face: The four eye corners and the two nostrils. This technique starts with the assumption that all model points lie in a plane parallel to the image plane (corresponds to a orthographic projection of the model into the image plane plus a scaling). Then, by iteration, the algorithm adjusts the model points until their projections into the image plane coincide with the observed image points. The pose of the 3D model is obtained by solving iteratively the following linear system of equations:

$$\begin{cases} M_i \cdot \frac{f}{Z_0} i = x_i(1 + \epsilon_i) - x_0 \\ M_i \cdot \frac{f}{Z_0} j = y_i(1 + \epsilon_i) - y_0 \end{cases}$$

M_i is the position of object point i , i and j are the two first base vectors of the camera coordinate system in object coordinates, f is the focal length and Z_0 is the distance of the object origin from the camera. i , j , and Z_0 are the unknown quantities to be determined. x_i, y_i is the scaled orthographic projection of the model point i , x_0, y_0 is the origin of the model in the same plane, ϵ_i is a correction term due to the depth of the

model point, ϵ_i is the parameter that is adjusted in each iteration until the algorithm converges. This algorithm is very stable, also with measurement errors, and it converges in just a few iterations.

4.5 Errors

If the recognition module has failed to identify eyes or nostrils on a given frame, we simply ignore that frame during the model creation process. The recognition module marks the inner and outer corners of both eyes, as well as the center of the nostrils. The location of the nostrils is very reliable and robust. We are able to derive their position with sub-pixel accuracy by applying low-pass filtering on their trajectories. The location of the eye corners is less reliable because their positions change slightly during closures. We ignore frames on which the eyes are closed. The errors in the filtered positions of these feature points are typically less than one pixel. A study of the errors in the pose resulting from errors of the recognition is shown on table 1. All possible combinations of recognition errors are calculated for a given perturbation (with 6 points and 9 possible errors, $all 9^6 = 531441$ poses have been computed).

	0.5	1.0	1.5	2.0	1.0 (AVG)
X-angle [deg]	1.6	3.3	5.0	6.9	0.7
Y-angle [deg]	1.4	2.8	4.3	5.8	0.67
Z-angle [deg]	0.6	1.2	1.9	2.6	0.27
Z-pos [mm]	8.9	18.3	27.4	36.6	5.6

Table 1: The values shown in the table are the maximum errors in the calculated pose (x,y,z angles in degrees and distance to camera in mm) for perturbations of the measured feature points by: 0.5, 1, 1.5 and 2 pixels. The last column shows an average error for a perturbation of 1 pixel. The subject was at a distance of 1m from the camera. The camera focal length was 15mm and its resolution 560x480 pixels.

5 SAMPLES OF FACIAL FEATURES

5.1 Unit

There are two choices in selecting the unit of the samples, either single images, or short sequences of images. Bregler et al. [13] use video sequences of triphones as the basic sample unit. This results in large databases but allows a semantically meaningful parameterization and requires fewer samples for synthesizing a new sequence. We use both units: Single frames, to keep the database size low and short sequences where they are clearly advantageous for the animation. For facial parts, we use mostly single images. Since the recognition module provides extensive information about the shape of facial features it is possible to parameterize them reliably. For cases where the appearance of the facial part cannot be properly described by the chosen parameters, e.g. a smiling mouth, we store short sequences labeled with their appearance. For the base face we also store short sequences of typical head movements.

5.2 Normalization

Before the image samples are entered into the database they are corrected in shape and scale to compensate for the different head orientations when they were recorded. From the recognition module the position and shape of facial parts as well as the pose

of the whole head are known (figure 3a). To extract facial parts from the images we first project the planes of the 3D model into the image plane (figure 3b). The projected planes then mark the extent of each facial part (figure 3c). These areas are "un-warped" into normalized bitmaps (figure 3d). Any information about the shape produced by the recognition module is also mapped into the normalized view and stored along the bitmap in a data-structure. For example, the recognition module produces the outline of the lips encoded as a sequence of points. All these points are mapped into the normalized plane before entering them into the database.

5.3 Parameterization

Once all samples of a face part are extracted from the video sequences and normalized, they need to be labeled and sorted in a way that they can be retrieved efficiently. To parameterize a facial part we choose some of the measurements produced by the recognition module. In figure 3e, for example, we parameterize the mouth with three parameters: The width (the distance between the two corner points), the y-position of the upper lip (the y-maximum of the outer lip contour) and the y-position of the lower lip (the y-minimum of the outer lip contour). Samples of other facial parts are parameterized in a similar way.

Beside geometric features, we also use parameters describing the appearance of a facial part. The filtering processes described in section 4.1 provide a convenient way of characterizing the texture of a sample. By filtering a bitmap with a band-pass filter and measuring the intensity in three or four frequency bands, we obtain a characterization of the texture that can be used to parameterize the samples. In this way, we can differentiate between samples that have the same geometrical dimensions, but a different visual appearance.

The space defined by the parameters of a face part is quantized at regular intervals. This creates an n -dimensional grid, where n is the number of parameters (figure 3f), and each grid point represents a particular shape. First the numbers of intervals on the axes of the grid are chosen, then all samples are scanned and the distribution of each parameter analyzed. Based on this information the exact positions of grid intervals are set.

5.4 Database

Searching through all samples, we now populate each grid point with the k closest candidate bitmaps. Three parameters govern the size t of the resulting database of samples: The number of parameters n and the number of intervals p on each axis of the parameter space and k , the number of samples kept at each grid point.

$$t = k \prod_{i=1}^n p_i$$

Having multiple samples per grid point, i.e. $k > 1$ is useful for several reasons. In the "debugging" phase of the database an operator can choose the best of a small set of automatically selected samples. Another reason to keep multiple samples is that such expressions as a smile or putting pressure on the lips produce visually different mouth shapes for one set of parameters. One could increase the dimensionality of the parameter space, yet this would increase the number of samples drastically. By selectively populating grid points with more than one sample one can cover such cases more efficiently.

There is a trade-off between the size of the database and the quality of the animation that it can generate. Reducing the number of parameters will decrease the precision with which a sample can be characterized and result in a poor selection of samples. Reducing the number of intervals means bigger differences between neighboring samples and therefore the need to synthesize more transition samples that are of lower visual quality. In our example the mouth shape is characterized with 3 parameters, dividing them into 4, 4 and 3 intervals, respectively, resulting in a database of 48 mouth samples. About 40 additional samples are necessary to store the remaining facial parts. Each sample is about 5KB (compressed using JPEG), resulting in about 1/2MB of storage. We also need short sequences for the base face totaling about 2MB (compressed using MPEG2). Hence we have a very compact database of little over 2MB that produces high-resolution (560x480 pixels) animations. By scaling down the resolution we can generate animations from a database of a few hundred kilobytes.

5.5 Errors

Up to this point, the construction of the database was done fully automatically. But since no recognition system is 100% accurate, some erroneous samples will be included in the database. Errors of the recognition module include alignment errors and selection errors. Alignment errors are due to errors in the position of features, producing misaligned samples. Selection errors happen when parameters are not measured accurately. It is also possible that, for a given image, the parameters chosen do not characterize the appearance with sufficient accuracy. For example, two lip shapes may have the same parameters, yet look different because in one image the speaker puts more pressure on the lips. Such effects are corrected by synthesizing short animation sequences and verifying visually that they look smooth.

We developed a graphical interface allowing an operator to browse through the database, correct angles and positions of any sample and select new samples from a list of candidates if the sample at one grid point is not adequate. The system also tells the user which phonemes are mapped to the currently selected mouth viseme. This is useful to avoid articulation problems in the animation. By looking at short animations one can verify the visual continuity of the samples in the database. With this tool an operator creates a database in less than an hour.

6 ANIMATION

6.1 Trajectories

Choosing n parameters to describe a sample creates an n -dimensional space of possible appearances. An animation produces a parametric function (or trajectory) through this space with time as parameter (figure 4).

$$\text{traj}(t) = \{p_0(t), p_1(t), \dots, p_n(t)\}$$

Figure 4b shows the resulting trajectory in the three dimensional space of mouth parameters for the utterance: "I bet that". All the parameter values are given in Table 4c. To create a video animation at 30 frames per second, the trajectory is sampled every 33.33 milliseconds. Then for each sample point the closest grid entry and its associated bitmap is chosen. The parameters describing feature shapes are chosen so that transitions between neighboring samples look smooth. This guarantees that the resulting animation is also visually smooth.

```
foreach substring in string{
  if (substring.length < minlength, and
      substring is a plosive)
    substring is enlarged using
    surrounding substrings;
  else if (substring.length < minlength)
    substring takes the value of
    surrounding substring;
}
```

Definition: substring = consecutive samples of one viseme

Table 2: Example of the rule based algorithm used to filter abrupt transitions arising from quantizing of a trajectory into a string of samples.

Nevertheless the string of samples is the result of quantization, and without some minor filtering, quantization errors might result in abrupt transitions or visible artifacts. We use a rule-based filtering algorithm to eliminate these artifacts. The example in Table 2 illustrates the kind of rules that are used.

6.2 Transitions

To smooth transitions between samples further, we synthesize transition samples between two existing samples by blending them together using the following equation:

$$\text{pix}_{i,j} = \alpha \cdot \text{pixa}_{i,j} + (1 - \alpha) \cdot \text{pixb}_{i,j}$$

$$\alpha = \frac{t - t_0}{t_1 - t_0} \quad t \in [t_0, t_1]$$

During the transition interval from t_0 to t_1 the resulting pixel pix is a blend of the corresponding pixels from sample a (pixa) and sample b (pixb). The number of samples that are used to create a transition varies depending on the sampling rate of the trajectory and the duration of the samples. When the database contains few samples, the visual difference between samples is larger and more sophisticated techniques such as morphing [5] provide better results. Morphing is, however, computationally more expensive and requires correspondence points. When the visual difference between samples is reasonably low, the simpler, cheaper blending technique is adequate. Figure 4d shows the sequence of mouth shapes selected from the database, plus the transition shapes, marked with a green T.

6.3 Mouth

We animate the mouth of the talking head model from a string of phonemes. Each phoneme is mapped to its visual equivalent, a viseme (mouth sample). To account for coarticulation¹ we use a model described in [22].

Instead of directly mapping a phoneme to a viseme, we derive each parameter of a viseme v_p from a sequence of phonemes, where each phoneme has a target value v_{p0} and a decay function $g(t)$. The decay function is an exponential function describing the 'influence' a particular parameter has on its neighbors. The value of k is the span over which coarticulation is considered and corresponds to about 300 milliseconds. $v_{p,t}$

¹ Coarticulation refers to the effect that the lip shape is determined not only by the phoneme uttered at a time but also by previous and subsequent phonemes.

the value of parameter p at time t , defines a trajectory in the parameter space of the mouth shapes (figure 4b).

$$v_{p,j} = \frac{\sum_{j=0}^{t-1} v_{p0,j} \cdot g_p(t-j)}{\sum_{j=0}^{t-1} g_p(t-j)}$$

6.4 Other facial parts

We handle the animation of other facial parts using a model similar to the one developed for the MPEG4 facial animation subsystem [7]. Special markers are put in the text to control amplitude, length, onset and offset of facial animations. This is an easy way to provide synchronization of conversational cues, such as eye and eyebrow movements, eye blinks or head movements that accompany the spoken text.

6.5 Rendering

A frame of the final animation can be generated when bitmaps of all the face parts have been retrieved from the database. The bitmap of the base face is first copied into the frame buffer, then the bitmaps of face parts are projected onto the base face using the 3D model and the pose. At the moment we consider only a limited range of rotation angles of $\pm 15^\circ$ so that there is no need for hidden surface removal. To avoid any artifacts from overlaying bitmaps, we use gradual blending or "feathering" masks. These masks are created by ramping up a blending value from the edges towards the center. These operations are implemented using basic OpenGL calls and the whole frame is rendered with just a few texture-map operations, which makes it possible to render the talking head in real time on a low cost PC.

6.6 Text-To-Speech synthesizer

The whole animation is driven by the output of the text-to-speech synthesizer (TTS). Starting from ascii text input plus some annotation controlling the intonation, the TTS produces a sound file. In addition it also outputs a phonetic transcription. This includes precise timing information for each phoneme plus some information about the stress. The animation module translates this information into a sequence of visemes (see figure 4c). The stress information can be used to guide facial expressions and head movements.

Since we are striving for natural appearance of our face, we were searching for a TTS that sounds natural. Most TTS today produce speech that has a distinctly robot-like sound. Only very recent progress in speech synthesizer technology has produced speech that can be considered naturally sounding [2].

7 RESULTS AND DISCUSSION

We have produced short videos from two different head models, one female and one male. Figure 4 shows a few frames extracted from such a video. It is, of course, impossible to judge the quality of an animation from still pictures, this only shows that statically these frames look natural with no noticeable artifacts. To obtain feedback on the quality of these sequences, we have made informal tests with dozens of people. All tests were done with short clips, without integrating them into an application or trying to make them particularly entertaining (such

as telling a joke). In such a setting the viewers concentrate fully on the talking head and notice any artifacts. While reactions are mostly positive, some viewers criticize the lip synchronization and the articulation - often over-articulation. Occasionally blending artifacts at the teeth and the jaw are visible.

A formal test was done, to determine whether a talking head could improve intelligibility of spoken text in a noisy environment [1]. Two head models were tested, one 3D model, with and without texture maps of a real person, and this sample-based head. All head models improved intelligibility significantly and by about the same amount. These tests were done with an older version of the sample-based talking head and all heads used an older TTS [8], which has more of a robot-like voice. Subjective tests indicated that users dislike the clash between a naturally looking face and an unnatural voice (in contrast, a synthetic looking head with a natural voice seems to be perfectly acceptable). Therefore, in some of the tests the bare 3D model scored higher in 'being liked' than either the sample-based head or the 3D head with a person's texture map. The new TTS [2], sounds very natural and fits well the appearance of the sample-based head. In recent tests there have never been any complaints about a mismatch between voice and face.

A long-term goal is to produce animation snippets that cannot be distinguished from a real person or at least to make the animations look so good that a viewer accepts them as a replacement for video clips of a person. In order to make a talking head a valuable addition to an application, it is not only its appearance that must be of very high quality. To keep a viewer pleased, the talking head must have a wide repertoire of behaviors, blending discreetly into the flow of action of the surrounding application.

7.1 Coarticulation

The coarticulation model was originally developed to study speech production. The resulting mouth shapes tend to over-articulate certain phonemes, giving some of the animations an unnatural look. It is a generic model adapted to a particular subject by adjusting many parameters. Yet with such a synthetic model it is difficult to capture the details of how a person is articulating speech. We are in the process of converting this generic model to a data-driven model. To accomplish this we record videos of commonly spoken sequences of diphones, triphones and quadriphones. We then extract and normalize the trajectory of each lip parameter during articulation of the phonemes. Even though the number of these trajectories can be large, the size of each trajectory amounts to only a few hundred bytes, therefore resulting in a compact database. To synthesize new articulations of speech, the appropriate phoneme sequences are identified in the coarticulation database and are concatenated.

7.2 Model

The simple 3D model we currently use covers a limited range of views. This is because it approximates facial parts with only a few planes, resulting in visible artifacts when the head is rotated beyond about $\pm 15^\circ$ of the original sample's angle. To circumvent this limitation we plan to augment the model with new sets of samples that are extracted under different views. In this way a wider range of possible views can be covered by switching between sets of samples depending on the pose of the base head.

Emotional expressions are generated mostly through animations of the upper part of the face or when there is no

talking. More samples of mouth shapes will be added where the person is, for example, smiling while talking.

8 CONCLUSIONS

We have presented a novel way to create head models that can be used to generate photo-realistic talking-head animations. Using image samples captured while a subject was speaking preserves the original appearance. Image analysis techniques make it possible to compute the pose of the head and measure facial parts on tens of thousands of video frames, resulting in a rich, yet compact database of samples. A simple 3D model of the head and facial parts enables perspective projection of the samples onto a base head in a given pose, allowing head movements. The results are lively animations with a pleasing appearance that resemble closely a real person. This system looks promising for generating talking heads that can enliven computer-user interfaces as well as future encounters among avatars in cyberspace.

Recent advances in accuracy and robustness of face recognition systems make the approach described here feasible. Combining machine vision with computer graphics is an idea that is receiving increasing attention recently [9]. Using photographs as parts of computer graphics is an old tradition. Yet as long as photographs had to be segmented manually, this approach was very costly. As image analysis algorithms mature, more and more general scenes can be segmented automatically, opening a whole new world of possibilities for synthesizing photorealistic scenes.

Acknowledgements

We thank J. Ostermann and I. Pandzic for including our head models in their tests of user interfaces, as well as for many useful discussions. The text-to-speech synthesizer used here was developed by J. Schröter, M. Beutnagel, A. Conkie, Y. Stylianou, and A. Syrdal who provided access to timing and prosodic information and made the system available at an early stage of the development.

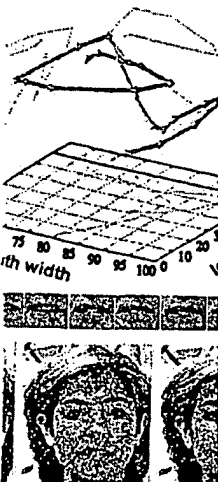
References

- [1] Pandzic, I., Ostermann, J., Millen, D., *Synthetic Faces: What are they good for?*, submitted to Visual Computer, 1999.
- [2] Stylianou, Y., *Concatenative speech synthesis using a Harmonic plus Noise Model*, Third ESCA Speech Synthesis Workshop, pp.261-266, Australia, Nov. 1998.
- [3] Guenter, B., Grimm, C., Wood, D., Malvar, H., Pighin, F., *Making Faces*, Proc. of SIGGRAPH'98, pp.55-66, ACM SIGGRAPH, July 1998.
- [4] Pighin, F., Hecker, J., Lichinski, D., Szeliski, R., Salesin, D.H., *Synthesizing Realistic Facial Expressions from Photographs*, Proc. of SIGGRAPH'98, pp.75-84, ACM SIGGRAPH, July 1998.
- [5] Ezzat, T., Poggio, T., *MikeTalk: A Talking Facial Display Based On Morphing Visemes*, Proc. of Computer Animation, IEEE Computer Society, pp.96-102, June 1998.
- [6] Cosatto, E., Graf, H.P., *Sample-Based Synthesis of Photo-Realistic Talking-Heads*, Proc. of Computer Animation, IEEE Computer Society, pp.103-110, June 1998.
- [7] Ostermann, J., *Animation of synthetic faces in MPEG-4*, Proc. of Computer Animation, IEEE Computer Society, pp.49-55, June 1998.
- [8] Olive, J., Van Santen, J., Moebius, B., Shih, C., *Synthesis*, chap 7, pp.191-228, in *Multilingual Text-to-Speech Synthesis*, Sproat, R., editor, Kluwer, 1998.
- [9] Lengyel, J., *The Convergence of Graphics and Vision*, IEEE Computer, vol. 31, n°7, pp.46-53, July 1998.
- [10] Potamianos, G., Graf, H.P., Cosatto, E., *An Image Transform Approach for HMM Based Automatic Lip Reading*, Proc. IEEE Int. Conf. on Image Processing, Vol.III pp.173-177, 1998.
- [11] Parke, F.I., Waters, K., *Computer Facial Animation*, A.K. Peters, Wellesley, Massachusetts, 1997.
- [12] Escher, M., Magnenat-Thalmann, N., *Automatic 3D Cloning and Real-Time Animation of a Human Face*, Proc. of Computer Animation, pp.58 - 66, IEEE Computer Society 1997.
- [13] Bregler, C., Covell, M., Slaney, M., *Video Rewrite: Driving Visual Speech with Audio*, Proc. SIGGRAPH'97, pp.353-360, ACM SIGGRAPH, July 1997.
- [14] DeCarlo, D., Metaxas, D., *Optical Flow Constraints, on Deformable Models with Applications to Human Face Shape and Motion Estimation*, Proc. CVPR, pp.231-238, 1996.
- [15] Beymer, D., Poggio, T., *Image Representation for Visual Learning*, Science, vol. 272, pp.1905-1909, 28 June 1996.
- [16] Seitz, S.M., Dyer, C.R., *View Morphing*, Proc. SIGGRAPH'96, pp.21-30, ACM SIGGRAPH, July 1996.
- [17] Bichsel, M., *Automatic Interpolation and Recognition of Faces by Morphing*, Proc. 2nd Int. Conf. on Automatic Face and Gesture Recognition, pp.128-135, IEEE CS Press, 1996.
- [18] Graf, H.P., Cosatto, E., Potamianos, G., *Robust Recognition of Faces and Facial Features with a Multi-Modal System*, Proc. IEEE Int. Conf. on Systems, Man and Cybernetics, pp.2034-2039, 1997.
- [19] Lee, Y., Terzopoulos, D., Waters, K., *Realistic Modeling for Facial Animation*, Proc. SIGGRAPH'95, pp.55-62, ACM SIGGRAPH, July 1995.
- [20] Chen, E.S., *QuickTime® VR - An Image-Based Approach to Virtual Environment Navigation*, Proc. SIGGRAPH'95, pp.29-38, ACM SIGGRAPH, July 1995.
- [21] Oberkampf, D., Dementhon, D., Davis, L., *Iterative Pose Estimation Using Coplanar Feature Points*, Internal Report, CVL, CAR-TR-677, University of Maryland, July 1993.
- [22] Cohen, M.M., Massaro, D.W., *Modelling Coarticulation in Synthetic Visual Speech*, in: *Models and Techniques in Computer Animation*, M. Magnenat-Thalmann and D. Thalmann (eds.), Springer Verlag, Tokyo, 1993.
- [23] Wolberg, G., *Digital Image Warping*, IEEE Computer Society Press, 1990.
- [24] Faiglin, G., *The Artist's Complete Guide to Facial Expression*, Watson-Guptill, New-York, 1990.
- [25] Ekman, P., Friesen, W., *Facial Action Coding System: A Technique for the Measurement of Facial Movement*, Consulting Psychologists Press, Palo Alto, CA, 1978.



Figure 1: Model of the talking-head: The base face (a) and the normalized facial parts (b). The head pose in the base face model of the head are used to project the facial parts (c). (d) shows a strongly rotated head pose to illustrate the 3D nature of the parts; this pose is rotated too much to be blended into a base face without artifacts. In (e) the projected facial parts are on base face (with the 3D wireframe removed (f)). The same facial parts are projected onto a different base face (g).

(a)
 wonderful, I bet that sound:



ation process: From ascii text (a) for each frame, parameters that meter values in % of their respective are then quantized to index ure closures and avoid jerky m-tered grid indices defines a string ne base face (c).

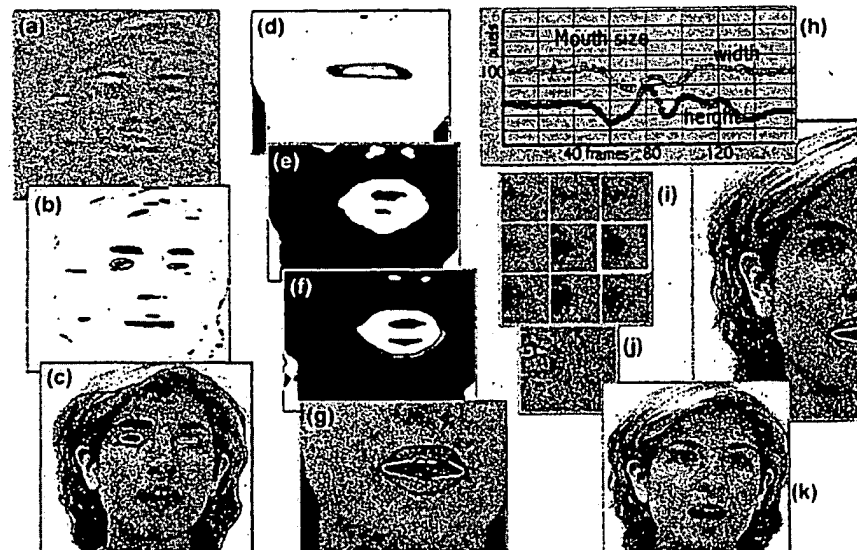


Figure 2: Recognition process: A frame is first filtered using a combination of bandpass and morphological filters. The image is thresholded, connected blobs are extracted, and their shapes analyzed (b). Using a model of the head, the shape is scored and the best scoring combination is kept. This marks the positions of the main facial features (c). Knowing the mouth, a color analysis is performed to find the outline of the lips (d) (e) (f). Then the lips are measured (h). (i) shows convolution kernels used to find the exact position of the mouth corner. One of them is selected (yellow box in (i)) and the mouth corner (filtered first) (j) is convolved with this kernel. The result of this operation provides a very precise location of the mouth corner (k). A similar analysis is performed for the eye corners and the corners of the eye brows (n).

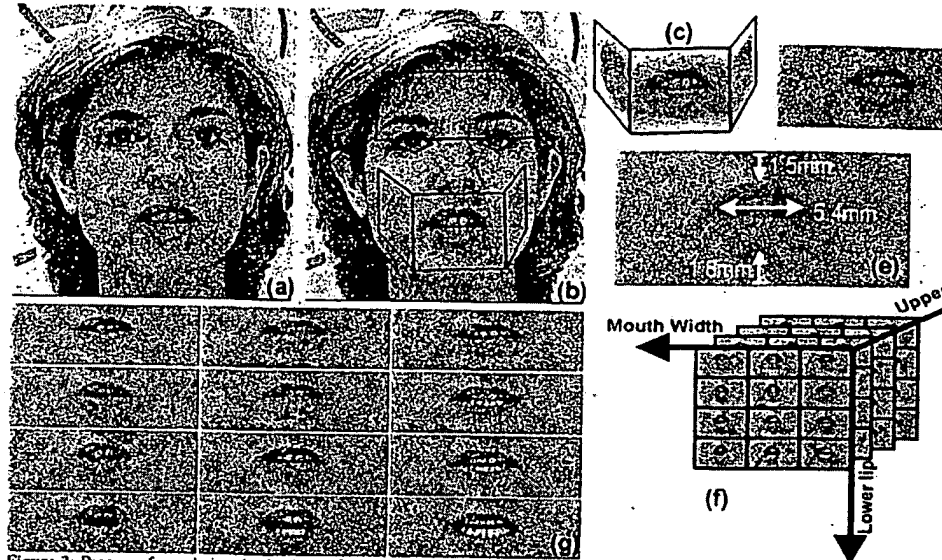


Figure 3: Process of populating the database of samples: From the recognized eyes and nostrils (a), the pose of the head is calculated. The 3D model is used to mark the areas of the facial parts (c), which are extracted and normalized (un-projected) (d). Feature measured on the sample for parameterization (e) and the space of samples is populated (f). The samples in (g) show a subspace where upper-lip parameter is constant.

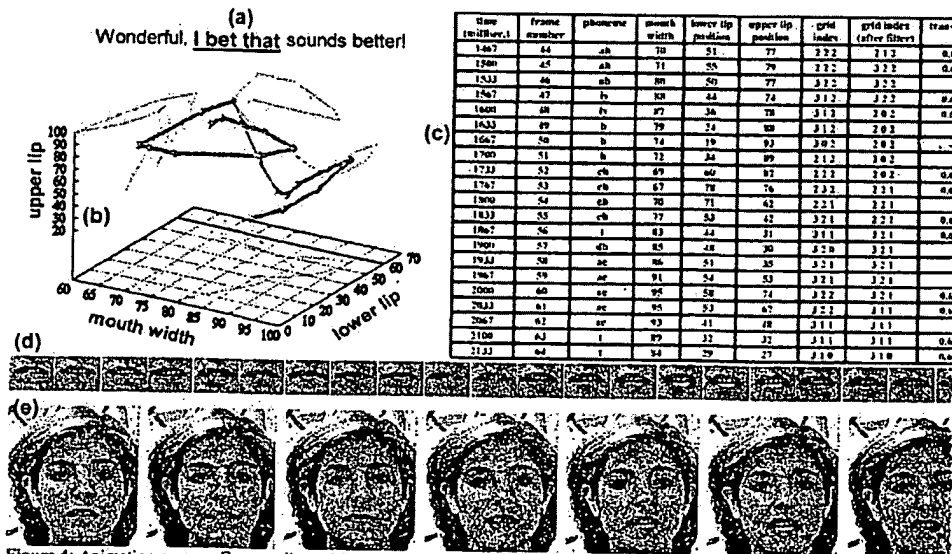


Figure 4: Animation process: From ascii text (a), the text-to-speech synthesizer provides a string of phonemes and their timing (c: 1st 3rd col.). Then for each frame, parameters that describe the mouth shapes are computed using coarticulation: (c), columns 4, 5, and 6 show the parameter values in % of their respective range. These values define the trajectory in the 3 dimensional space of the parameters (b). They are then quantized to index the database grid (in this case, a 4 by 4 by 3 grid) (c: 7th col.). This string of grid indices is filtered to ensure closures and avoid jerky movements (c: 8th col.). Transitions are inserted to smooth out the animation (c: 9th col.). The string of filtered grid indices defines a string of mouth bitmaps (d) (green T indicate transition bitmaps). Finally the mouth shapes combined with the base face (e).

Application/Control Number: 10/662,550
Art Unit: 2628

Docket No.: 2000-0042-CON

EXHIBIT #2

DATE:

4/4/2000

TO:

Ann Taylor
Outside Counsel Coordinator

RE:

IDS No. 2000-0042

Case Name: _____

DIRECT MANAGED CASE

☒ This submission was approved for filing on 3/17/2000 by the
multimedia IPR team.

☒ It is recommended that this submission be assigned to the firm of
off Wendy Koba If possible, _____ should
write the application.

☐ This submission is already assigned to the firm of _____.

☐ This application is already/should be (circle one) assigned to the firm of _____
and the status of the U.S. Case is:

- ☐ Authorized
- ☐ Provisional filed on _____
- ☐ Pending
- ☐ Amendment due out on _____
- ☐ Patented
- ☐ Response to Office Action in Foreign Country (_____) due out on _____
- ☐ Other: _____

Please note that the Case Folders are:

- ☒ Attached herewith
- ☒ U.S. ☐ Foreign (_____)
- ☐ Already in Middletown
- ☐ MIA (missing)
- ☐ Other: _____

SPECIAL COMMENTS:

TAR/07
Attorney/secy initials

4/27/99 amf

Application/Control Number: 10/662,550
Art Unit: 2628

Docket No.: 2000-0042-CON

EXHIBIT #3

Application/Control Number: 10/662,550
Art Unit: 2628

Docket No.: 2000-0042-CON

TAR

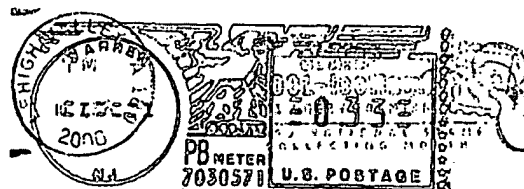


Disclosure No. 2000-0042 Our Ref. No. ATT-0186
Assigned Attorney: Wendy W. Koba, Esq.
Office Address: PO Box 556
Springtown, PA 18081
Phone No. 610-346-7112
Expected Filing Date 7/10/00

Application/Control Number: 10/662,550
Art Unit: 2628

Docket No.: 2000-0042-CON

AT&T PATENT	
APR 12 2000	
DOCKETED	<i>me</i>
DIRECT MAIL	
NOT DOCKETED	
PREVIOUSLY DOCKETED	
PREVIOUSLY ASSIGNED	



AT&T CORP.
P.O. BOX 4110
MIDDLETOWN, NEW JERSEY 07748
U.S.A.

07748-4110

